Grenze International Journal of Engineering and Technology, January Issue

Smart Card Fraud Detection using Ensemble Methods in Machine Learning

Nagula Varshitha[1], Donthula Richitha[2], Mandala Thanuj[3], and Rama Devi Burri[4]
[1-4]Institute of Aeronautical Engineering, Hyderabad, India
Email: varshitha.nagula99@gmail.com, richithadonthula1@gmail.com, thanujmandala123@gmail.com, ramaburri5@gmail.com

***Abstract*— This study examines the growing use of cards, online payments and the associated risk of fraud, which has led to significant financial losses globally. Financial institutions like banks need to detect fraudulent transactions effectively, prompting an analysis of different machine learning classifiers. The findings reveal that the Decision Tree algorithm outperformed Random Forest, achieving higher accuracy and superior fraud detection rates. Random Forest, on the other hand, struggled to handle the highly imbalance in the dataset, leading to notably lower accuracy. To overcome these limitations and enhance fraud detection capabilities, a hybrid model was developed, combining the strengths of both Decision Tree and Random Forest. This ensemble method achieved enhanced accuracy and robustness by leveraging the interpretability of Decision Trees and the feature-selection capabilities of Random Forest. The paper also highlights the importance of using machine learning to prevent fraudulent transactions and improve transaction security.**

***Index Terms*— Machine Learning, Smart Card, Cyber Security, Fraud Detection, Ensemble Method.**

I. INTRODUCTION

A smart card is a payment card issued by banks that allows users to make cashless transactions for goods and services [1]. It operates on a preset credit limit and can be used both online and in-store. Smart cards offer convenience but can lead to debt if not managed properly [9]. Smart card fraud has unauthorized use of credit card data, usually obtained from hacking or phishing, to make illegal transactions [8]. This form of fraud can result in substantial monetary losses for both individuals and businesses. Due to the difficulty in distinguishing fraudulent transactions from legitimate ones, credit card companies face challenges in effectively detecting fraud. Smart card fraud has become an important concern with the increasing reliance on digital payment methods. Smart cards come in various forms, such as revolving, charge, instalment, and non-instalment cards, each designed to cater to different financial needs [7].

Unfortunately, these cards are highly susceptible to fraudulent activities, often occurring during online transactions where only the card information is required, making it easy for criminals to exploit [3]. The rapid evolution of fraud techniques, such as creating counterfeit cards or manipulating cardholder information, poses a considerable challenge to both cardholders and financial institutions [4]. To address this, machine learning (ML) offers a robust solution for detecting and preventing smart card fraud. Machine learning models can process large volumes of transaction data, detecting patterns and irregularities that could signal fraudulent activities.

Grenze ID: 01.GIJET.11.1.262

By employing ensemble methods, such as Random Forest (RF) and AdaBoost, the detection accuracy can be significantly improved [12]. These models work by classifying transactions as either genuine or fraudulent based on historical information, allowing for real-time detection and response. A smart card is a payment card issued by banks that allows users to make cashless transactions for goods and services [10].
It operates on a preset credit limit and can be used both online and in-store. Smart cards offer convenience but can lead to debt if not managed properly.
Smart card fraud has unauthorized use of credit card data, usually obtained from hacking or phishing, to make illegal transactions.
This form of fraud can result in substantial monetary losses for both individuals and businesses. Due to the difficulty in distinguishing fraudulent transactions from legitimate ones, credit card companies face challenges in effectively detecting fraud [2].
Smart card fraud has become an important concern with the increasing reliance on digital payment methods. Smart cards come in various forms, such as revolving, charge, instalment, and non-instalment cards, each designed to cater to different financial needs.
Unfortunately, these cards are highly susceptible to fraudulent activities, often occurring during online transactions where only the card information is required, making it easy for criminals to exploit [11].
The rapid evolution of fraud techniques, such as creating counterfeit cards or manipulating cardholder information, poses a considerable challenge to both cardholders and financial institutions.
To address this, machine learning (ML) offers a robust solution for detecting and preventing smart card fraud. Machine learning models can process large volumes of transaction data, detecting patterns and irregularities that could signal fraudulent activities [5].
By employing ensemble methods, such as Random Forest (RF) and AdaBoost, the detection accuracy can be significantly improved [6]. These models work by classifying transactions as either genuine or fraudulent based on historical information, allowing for real-time detection and response.

II. Related Work

Ashawa et al. (2024) proposed a comprehensive fraud detection model that integrates multiple classifiers like Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN) within ensemble frameworks such as Bagging and Boosting. The approach tackles class imbalance using SMOTE and under-sampling, significantly improving detection accuracy. Their system demonstrated superior performance in identifying fraudulent transactions, emphasizing ensemble methods as a robust solution to minimize false positives [1].
Choi and Lee (2024) developed a hybrid machine learning model combining supervised and unsupervised learning approaches for detecting mobile payment fraud. The system incorporated ensemble techniques to enhance scalability and accuracy, efficiently handling large datasets. Feature selection and data sampling strategies improved processing time and model precision. Validation using F-measure and ROC curves highlighted its effectiveness for real-world applications [2].
Sharma et al. (2023) emphasized the use of Random Forest and Gradient Boosting algorithms in detecting anomalies in financial transaction data. Their ensemble approach focused on scalability and real-time processing, leveraging advanced feature engineering techniques to minimize latency while maintaining high accuracy. The study illustrated the potential of boosting-based ensembles in improving fraud detection rates in dynamic environments [3].
Bhandari et al. (2022) introduced an anomaly detection framework that combines Isolation Forest and one-class SVM algorithms within an ensemble system.
Their model efficiently identified outliers associated with fraudulent activities, particularly in highly imbalanced datasets. The ensemble structure was shown to enhance generalization and reduce overfitting, making it suitable for diverse fraud detection scenario [4].
Rahman et al. (2020) explored Bagging and Decision Trees as ensemble techniques for fraud detection, focusing on improving precision and recall. Their study highlighted the value of combining weak learners to create a robust detection system. The results underscored the significance of optimizing ensemble configurations to enhance predictive accuracy for specific transaction datasets [5].
Chen et al. (2020) proposed an ensemble learning system integrating Decision Trees, Gradient Boosting Machines, and Voting Classifiers for fraud detection. Using data augmentation to address the rarity of fraud cases, the model achieved superior sensitivity and specificity. The study demonstrated the potential of ensemble approaches to handle challenges like class imbalance and data sparsity effectively [6].

III. METHODOLOGY

In proposed methodology, ensemble random forest and decision tree algorithm are used for smart card fraud trans action classification from highly imbalanced datasets. Combining Random Forest and Decision Tree algorithms can enhance predictive performance by leveraging their complementary strengths. A Decision Tree is simple and intuitive, making decisions based on feature splits to classify or predict outcomes.
However, it can be prone to overfitting, especially with complex datasets. On the other hand, a Random Forest, which is an ensemble model of multiple decision trees, mitigates overfitting by averaging the predictions of many trees, thus providing more stable and accurate results. A common approach to combining these models is stacking, where both a Decision Tree and a Random Forest are trained separately, and their predictions are used as inputs to a final meta-model. This meta-model, often another Decision Tree or a different classifier, learns from the combined predictions to make the final decision. This strategy allows the strengths of the Random Forest in handling feature importance and variance reduction to complement the simplicity and interpretability of the Decision Tree, resulting in a robust and reliable predictive model.

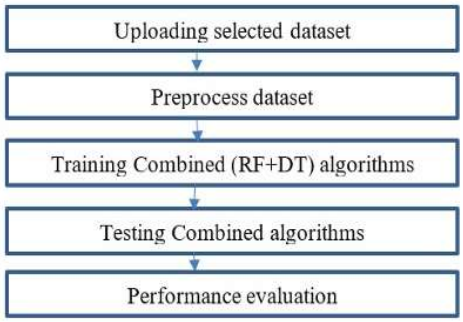

Fig. 1. Block Diagram of Proposed Methodology

A. Uploading the Selected Dataset

The dataset selected for this application is noticeably imbalanced, with a much larger number of transactions included in the training set compared to the testing set. The large training set provides the model with ample data to learn intricate patterns and relationships, which is especially useful in tasks like fraud detection, where identifying uncommon behaviours requires a diverse set of examples. In contrast, the testing set contains a significantly smaller number of transactions, which may limit the effectiveness of evaluating the model's ability to generalize to new data. Additionally, the uneven distribution between the training and testing sets may obscure issues like overfitting or underfitting. To mitigate these concerns, strategies such as stratified sampling, cross-validation, or generating synthetic samples (e.g., SMOTE) can be applied. These approaches help balance the dataset and ensure it accurately represents all classes, enhancing the model's performance and robustness.

B. Preprocessing the Selected Dataset

During the preprocessing phase, missing values in the dataset are systematically addressed to maintain data quality and ensure it is suitable for analysis. This step is crucial because missing data can hinder the model's ability to identify meaningful patterns. Depending on the nature of the dataset, various imputation methods can be used, such as replacing missing values with the mean, median, or mode, or employing advanced techniques like k-Nearest Neighbours (k-NN) imputation. In addition to handling missing values, preprocessing involves formatting the data to meet the requirements of the model. These steps are essential for reducing inconsistencies and ensuring the dataset is ready for analysis. Once the data is preprocessed, it is divided into two parts: 80% is used for training the model, while 20% is set aside for testing. This division provides the model with a large portion of data to learn patterns and relationships while retaining enough unseen data to evaluate its performance. The commonly used 80-20 split strikes a balance between training efficiency and reliable testing. When splitting the data, it is often necessary to preserve the class distribution, especially in imbalanced datasets, which can be achieved through stratified sampling. These measures ensure that the model is trained effectively and evaluated fairly, improving its ability to generalize to new data.

C. Training Machine Learning Algorithms

The dataset, after being split, is used to train multiple ma chine learning classifiers that utilize different methodologies for learning patterns. Supervised classifiers, which depend on labelled data, are designed to map

inputs to specific outputs by learning patterns from the training data. Algorithms such as Decision Trees, Random Forests, and Support Vector Machines (SVMs) are examples of this approach. In contrast, unsupervised classifiers work without labelled data, focusing on uncovering structures or patterns within the dataset. After training, both types of classifiers are evaluated on the testing set to assess their prediction accuracy. The evaluation results show that unsupervised classifiers achieve higher prediction accuracy compared to supervised ones. This outcome could be attributed to the characteristics of the dataset or the specific problem being addressed.

D. Testing Machine Learning Algorithms

Once the models have been trained, they are tested and used to make predictions on the testing dataset. This stage is essential for determining how well the models can generalize to new, unseen data. During testing, both supervised and unsupervised classifiers are applied to the test set, and their predictions are recorded. Supervised models, which are trained on labelled data, predict outcomes based on the patterns they have learned, while unsupervised models, without labelled data, detect patterns or clusters to make pre dictions. To measure the performance of these classifiers, their prediction accuracy is assessed using various performance metrics. By comparing the performance of supervised and unsupervised classifiers, one can determine which approach is more suitable for the specific problem. This evaluation is important for selecting the most effective model based on its ability to balance prediction accuracy with reliability, ultimately guiding the choice of the best classifier for practical use.

E. Performance Evaluation

Using above 5 steps proposed method can be designed which shows superior performance of proposed model over state of art technique. The dataset in this application is highly imbalanced dataset which contains normal trans actions in lakhs and fraud transactions in only hundreds. Proposed model even works with highly imbalanced dataset using machine learning techniques. Data cleaning and data splitting operations are performed as preprocessing operations for input dataset to make dataset in standard format. Machine learning classifiers are trained using training 80% of the samples and testing is performed on 20% of the dataset samples. Performance evaluation is applied at the last using balanced accuracy, precision, recall, etc. Performance analysis parameters shows that unsupervised ML techniques work superior than supervised ML techniques.

IV. Result Analysis

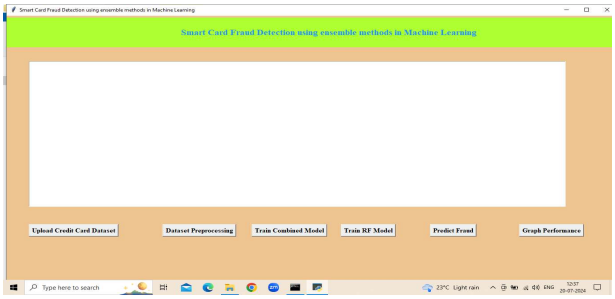

Fig.2 Graphical User Interface

After clicking on run.bat file, this Graphical User Interface will open as shown in fig 2.

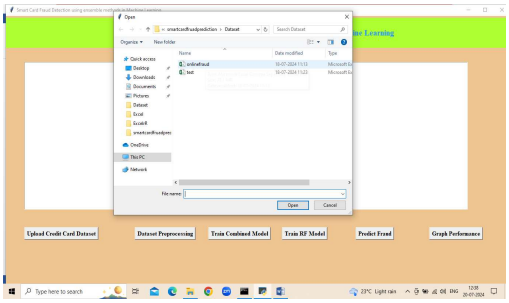

Fig.3 Uploading Dataset

First step is uploading dataset. After clicking on “Upload dataset” button, we have to select dataset file as shown in fig 3.

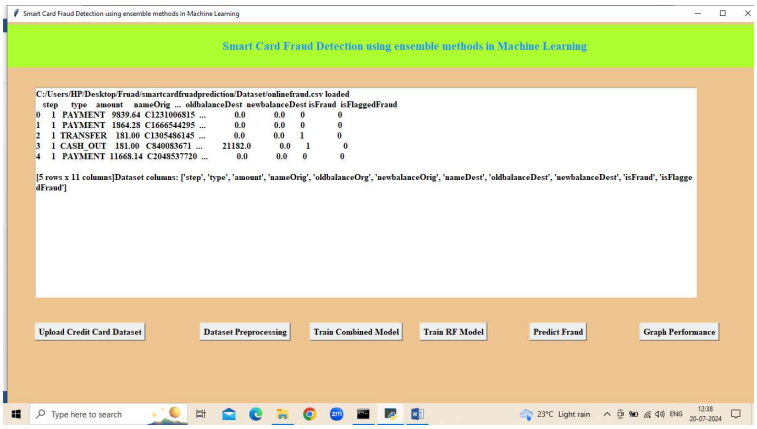

Fig.4 Dataset Loaded

Once dataset is loaded, it will display first five records of dataset as shown in fig 4.

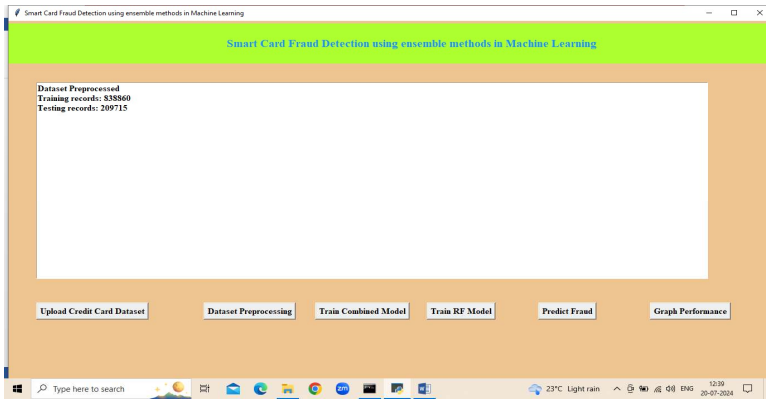

Fig.5 Dataset Preprocessing

Next step is data preprocessing. Here data will be preprocessed and it split the data into training and testing as shown in fig 5.

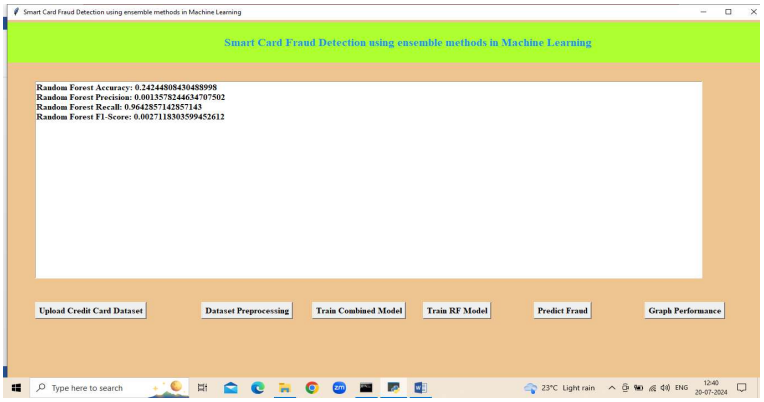

Fig.6 Train Random Forest Model

There is another individual Random Forest model will be trained. As shown in fig 6 it is getting accuracy 24% which is less compared to combined model.

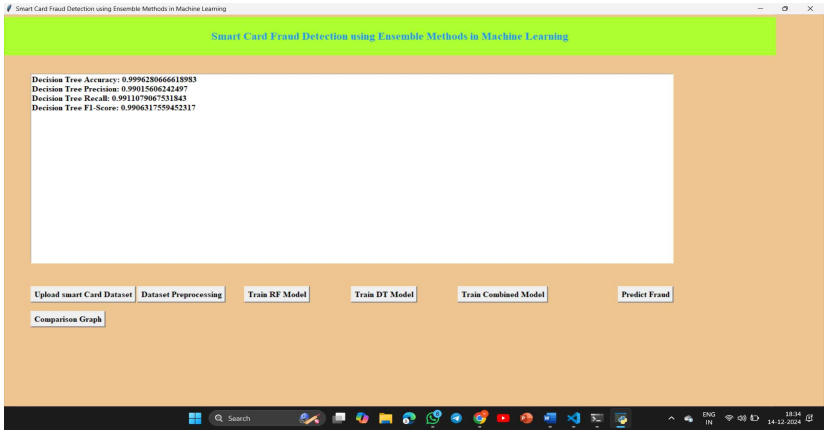

Fig.7 Train Decision Tree Model

There is another individual Decision Tree model will be trained. As shown in fig 7 it is getting accuracy 99.96% which is less compared to combined model.

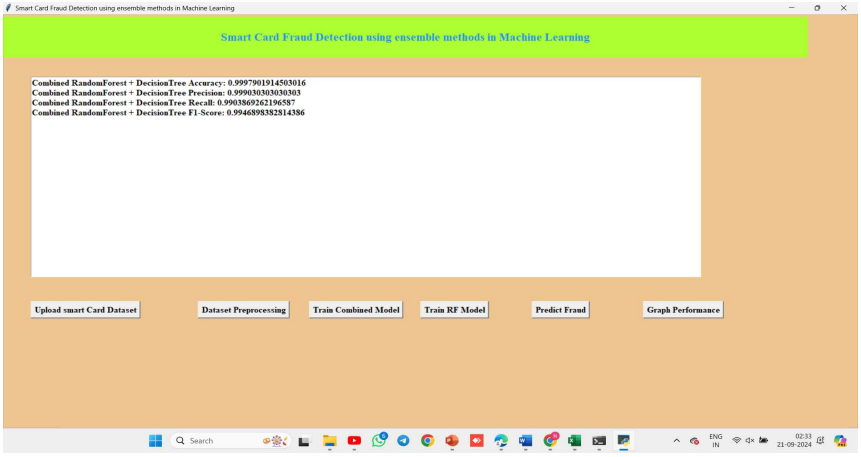

Fig.7 Train Combined RF+ DT model

We will train the combined Random Forest and Decision tree model. As shown in fig 7 it is getting 99.97% accuracy.

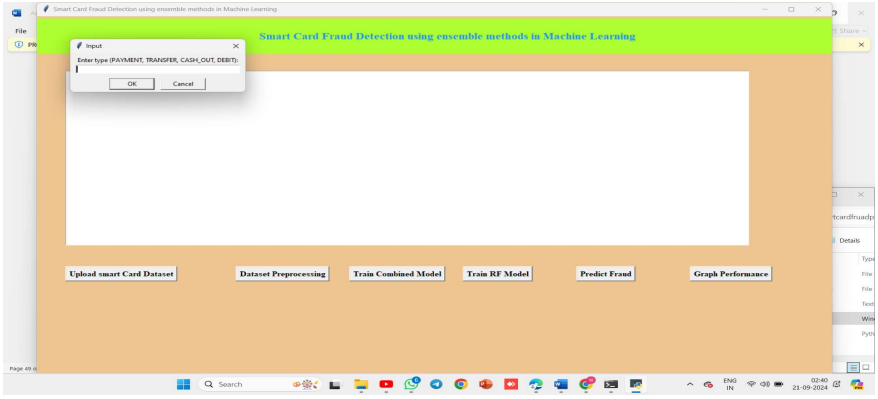

Fig.8 Providing input

After testing data using combined model, we fill the required data as shown in fig 8.

Fig.9 Fraud Prediction

After giving input values for prediction. As shown in fig 9 it will predict where the required details are fraud or non-fraud.

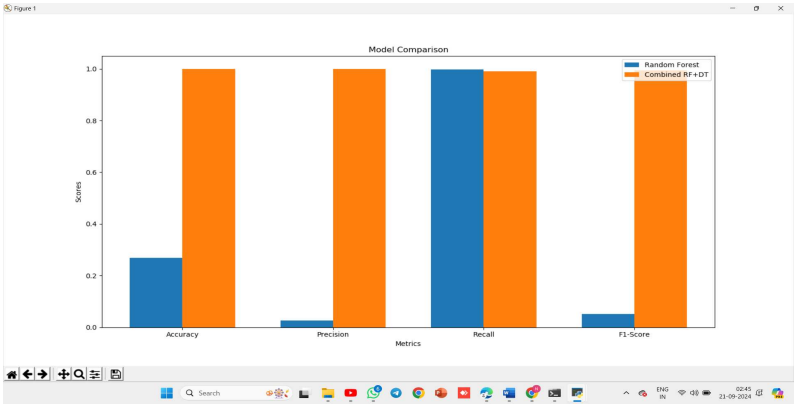

Fig.10 Algorithms Comparison Graph

Fig 10 is the graph comparison for random forest model and ensembled model.

TABLE I. COMPARATIVE PERFORMANCE OF MACHINE LEARNING MODELS

Model	Accuracy	F-Score	Recall	Precision
Random Forest	26.88	26.77	99.83	51.78
Decision Tree	99.96	99.01	99.11	99.06
Random Forest+Decision Tree	99.97	99.90	99.03	99.46

V. CONCLUSION

This study focuses on detecting smart card fraud using highly imbalanced datasets, employing machine learning algorithms such as Decision Tree and Random Forest classifiers. Performance evaluation was carried out using metrics like precision, recall, and balanced accuracy, which were derived from the confusion matrix. The Decision Tree algorithm was found to perform slightly better in terms of accuracy compared to Random Forest. However, to improve the overall performance, an ensemble approach combining both algorithms was introduced. The combined use of Decision Tree and Random Forest takes advantage of the unique strengths of both models. The Decision Tree provides efficiency and interpretability, while Random Forest offers improved generalization and reduces the risk of overfitting through its ensemble approach. Together, these models create a more robust fraud detection system, especially in dealing with imbalanced data. In conclusion, this study shows that ensemble methods significantly enhance fraud detection in smart card systems. By integrating Decision Tree and Random Forest, the proposed solution delivers higher accuracy and performance, making it a more reliable tool for real-world applications. Future research could focus on further optimizing the ensemble model or exploring additional machine learning techniques to improve adaptability and efficiency in various financial contexts.

REFERENCES

[1] A. Aftab, I. Shahzad, A. Sajid, M. Anwar, and N. Anwar, "Fraud Detection of Credit Cards Using Supervised Machine Learning Techniques.," Pakistan J. Emerg. Sci. Technol., p. 4, 2023, doi: 10.58619/pjest.v4i3.114

[2] Cherif, A. Badhib, H. Ammar, S. Alshehri, M. Kalkatawi, and A. Imine, "Credit card fraud detection in the era of disruptive technologies: A systematic review," J. King Saud Univ. - Comput. Inf. Sci., vol. 35, no. 1, pp. 145–174, 2023, [Online]. Available: https://doi.org/10.1016/j.jksuci.2022.11.008.

[3] J. Xia, "Credit Card Fraud Detection Based on Support Vector Machine.," Highlights Sci. Eng. Technol., vol. 23, pp. 93–97, 2022, doi: 10.54097/hset.v23i.3202.

[4] J. K. Afriyie et al., "A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions," Decis. Anal. J., vol. 6, p. 100163, 2023, [Online]. Available: https://doi.org/10.1016/j.dajour.2023.100163.

[5] J. Chen and K.-L. Lai, "Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert.," J. Artif. Intell. Capsul. Networks, vol. 3, pp. 101–112, 2021, doi: 10.36548/jaicn.2021.2.003.

[6] Dr. B. Rama Devi, Ms. K. Sri Harsha, Ms. Y. Himaja, Mr. B. Nagendra Babu "Fraud Detection on Smart Cards Using Machine Learning Algorithms.," Volume 83 Page Number: 2495 - 2501 Publication Issue: May - June 2020. Available: http://testmagzine.biz/index.php/testmagzine/article/view/7649.

[7] E. Ileberi, Y. Sun, and Z. Wang, "A machine learning based credit card fraud detection using the GA algorithm for feature selection.," J Big Data, vol. 9, p. 24, 2022, [Online]. Available: https://doi.org/10.1186/s40537-022-00573-8.

[8] A. Al-Faqeh, A. Zerguine, M. Al-Bulayhi, A. AlSleem, and A. Al-Rabiah, "Credit Card Fraud Detection via Integrated Account and Transaction Submodules," Arab. J. Sci. Eng., vol. 46, pp. 10023– 10031, 2021, [Online]. Available: https://doi.org/10.1007/s13369-021-05856-5

[9] O. Adebayo, T. Favour-Bethy, O. Owolafe, and O. Adebola, "Comparative Review of Credit Card Fraud Detection using Machine Learning and Concept Drift Techniques.," Int. J. Comput. Sci. Mob. Comput., vol. 12, pp. 24–48, 2023, doi: 10.47760/ijcsmc.2023.v12i07.004.

[10] E. Marazqah Btoush, X. Zhou, R. Gururajan, K. Chan, R. Genrich, and P. Sankaran, "A systematic review of literature on credit card cyber fraud detection using machine and deep learning.," Peer J Comput Sci., vol. 9, p. e1278, 2023, doi: 10.7717/peerj-cs.1278.

[11] E. Balagolla, W. Fernando, R. Rathnayake, M. Wijesekera, A. Senarathne, and K. Abeywardhana, "Credit Card Fraud Prevention Using Blockchain.," in 2021 6th International Conference for Convergence in Technology (I2CT), 2021, pp. 1–8, [Online].

[12] T. Vairam, S. Sarathambekai, S. Bhavadharani, A. Dharshini, N. Sri, and T. Sen, "Evaluation of Naïve Bayes and Voting Classifier Algorithm for Credit Card Fraud Detection.," in 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), 2022, vol. 1, pp. 602–608, [Online]. Available: https://doi.org/10.1109/ICACCS54159.2022.978496 8.